

## IN THE CLAIMS

23. (Currently Amended) A computer-implemented method comprising:

creating a first array for a first program source code file including a plurality of program elements, the first array containing program elements of a distinct program element type represented by any one of lines of functional program code, lines of program comments, or program code identifiers;

creating a second array for a second program source code file including the plurality of program elements, the second array containing program elements of the same program element type as the program elements in the first array;

comparing the program elements from the first array with the program elements from the second array to find similar program elements, the comparison being performed between individual program elements regardless of an order of code lines containing the program elements in the first array and the second array during the comparison, the comparison being independent of common sequences of program elements in the first array and the second array;

calculating a similarity number based on the similar program elements; and

presenting to a user an indication of plagiarism with respect to at least one of the first program source code file and the second program source file, wherein the indication of plagiarism is defined by ~~a larger~~ the similarity number ~~of program elements resulting from the comparison.~~

24. (Previously Presented) The method of claim 23 further comprising:

substituting each sequence of whitespace characters in each program element in the first array and the second array with a single space character.

25. (Previously Presented) The method of claim 23 wherein the comparison is insensitive to whether characters of the program elements are uppercase or lowercase.

26. (Previously Presented) The method of claim 23 wherein:

the program element type of the program elements in the first and second arrays is a line of functional program code; and

the method further comprises:

prior to comparing, eliminating from the first and second arrays lines of functional program code that consist entirely of programming keywords.

27. (Currently Amended) The method of claim 23 wherein:

the program element type of the program elements in the first and second arrays is a line of program comments; and

calculating a similarity number ~~comparing the program elements from the first array with the program elements from the second array~~ comprises finding a number of matching lines in the first and second arrays.

28. (Currently Amended) The method of claim 23 wherein:

the program element type of the program elements in the first and second arrays is a line of functional program code excluding functional program code consisting entirely of program language keywords; and

calculating a similarity number ~~comparing the program elements from the first array with the program elements from the second array~~ comprises finding a number of matching lines in the first and second arrays.

29. (Currently Amended) The method of claim 23 wherein:

the program element type of the program elements in the first and second arrays is a program code identifier; and

calculating a similarity number ~~comparing the program elements from the first array with the program elements from the second array~~ comprises finding a number of similar identifiers in the first and second arrays.

30. (Currently Amended) The method of claim 29 wherein the similar identifiers represent at least one of exact matches or partial matches.

31. (Cancelled)

32. (Previously Presented) The method of claim 31 wherein each partial match is a sequence of characters that can be found within an element of the first array and an element of the second array.

33. (Currently Amended) A computer-readable storage medium ~~having~~ storing executable instructions to cause a computer system to perform a method comprising:

creating a first array for a first program source code file including a plurality of program elements, the first array containing program elements of a distinct program element type represented by any one of lines of functional program code, lines of program comments, or program code identifiers;

creating a second array for a second program source code file including the plurality of program elements, the second array containing program elements of the same program element type as the program elements in the first array;

comparing the program elements from the first array with the program elements from the second array to find similar program elements, the comparison being performed between individual program elements regardless of an order of code lines containing the program elements in the first array and the second array during the comparison, the comparison being independent of common sequences of program elements in the first array and the second array; and

calculating a similarity number based on the similar program elements; and

presenting to a user an indication of plagiarism with respect to at least one of the first program source code file and the second program source file, wherein the indication of plagiarism is defined by ~~a larger~~ the similarity number ~~of similar program elements~~.

34. (Currently Amended) The computer-readable storage medium of claim 33 wherein the method further comprises:

substituting each sequence of whitespace characters in each program element in the first array and the second array with a single space character.

35. (Currently Amended) The computer-readable storage medium of claim 33 wherein:

the program element type of the program elements in the first and second arrays is a line of functional program code; and

the method further comprises:

prior to comparing, eliminating from the first and second arrays lines of functional program code that consist entirely of programming keywords.

36. (Currently Amended) The computer-readable storage medium of claim 33 wherein:

the program element type of the program elements in the first and second arrays is a line of program comments; and

calculating a similarity number ~~comparing the program elements from the first array with the program elements from the second array~~ comprises finding a number of matching lines in the first and second arrays.

37. (Currently Amended) The computer-readable storage medium of claim 33 wherein:

the program element type of the program elements in the first and second arrays is a line of functional program code excluding functional program code consisting entirely of program language keywords; and

calculating a similarity number ~~comparing the program elements from the first array with the program elements from the second array~~ comprises finding a number of matching lines in the first and second arrays.

38. (Currently Amended) The computer-readable storage medium of claim 33 wherein:

the program element type of the program elements in the first and second arrays is a program code identifier; and

calculating a similarity number ~~comparing the program elements from the first array with the program elements from the second array~~ comprises finding a number of similar identifiers in the first and second arrays.

39. (Currently Amended) The computer-readable storage medium of claim ~~33~~ 38 wherein the similar identifiers represent at least one of exact matches or partial matches.

40. (Currently Amended) The computer-readable storage medium of claim 39 wherein each partial match is a sequence of characters that can be found within an element of the first array and an element of the second array.

41. (Currently Amended) A computer-implemented apparatus comprising:

a computer; and

a source code matching program on the computer, the source code matching program comprising:

means for creating a first array for a first program source code file including a plurality of program elements, the first array containing program elements of a distinct program element type represented by any one of lines of functional program code, lines of program comments, or program code identifiers;

means for creating a second array for a second program source code file including the plurality of program elements, the second array containing program elements of the same program element type as the program elements in the first array;

means for comparing the program elements from the first array with the program elements from the second array to find similar program elements, the comparison being performed between individual program elements regardless of an order of code lines containing the program elements in the first array and the second array during the comparison, the

comparison being independent of common sequences of program elements in the first array and the second array;

calculating a similarity number based on the similar program elements; and

means for presenting to a user an indication of plagiarism with respect to at least one of the first program source code file and the second program source file, wherein the indication of plagiarism is defined by ~~a larger~~ the similarity ~~number of similar program elements~~.

42. (Previously Presented) The apparatus of claim 41 wherein:

the program element type of the program elements in the first and second arrays is a line of functional program code; and

the source code matching program further comprises:

means for eliminating, prior to comparing, from the first and second arrays lines of functional program code that consist entirely of programming keywords.

43. (Currently Amended) The apparatus of claim 41 wherein:

the program element type of the program elements in the first and second arrays is a line of program comments; and

the means for calculating a similarity number ~~comparing the program elements from the first array with the program elements from the second array~~ comprises means for finding a number of matching lines in the first and second arrays.

44. (Currently Amended) The apparatus of claim 41 wherein:

the program element type of the program elements in the first and second arrays is a line of functional program code excluding functional program code consisting entirely of program language keywords; and

the means for calculating a similarity number ~~comparing the program elements from the first array with the program elements from the second array~~ comprises means for finding a number of matching lines in the first and second arrays.

45. (Currently Amended) The apparatus of claim 41 wherein:

the program element type of the program elements in the first and second arrays is a program code identifier; and

the means for calculating a similarity number ~~comparing the program elements from the first array with the program elements from the second array~~ comprises means for finding a number of similar identifiers in the first and second arrays.

46. (Currently Amended) The apparatus of claim 45 wherein the similar identifiers represent at least one of exact matches or partial matches.

47. (Previously Presented) The apparatus of claim 46 wherein each partial match is a sequence of characters that can be found within an element of the first array and an element of the second array.

Please add the following new claims:

48. (New) The method of claim 23 wherein the similar program elements represent at least one of exact matches or partial matches.



49. (New) The computer-readable storage medium of claim 33 wherein the similar program elements represent at least one of exact matches or partial matches.

50. (New) The computer-readable storage medium of claim 33 wherein the comparison is insensitive to whether characters of the program elements are uppercase or lowercase.

51. (New) The apparatus of claim 41 wherein the similar program elements represent at least one of exact matches or partial matches.

52. (New) The apparatus of claim 41 further comprising means for substituting each sequence of whitespace characters in each program element in the first array and the second array with a single space character.

53. (New) The apparatus of claim 41 wherein the comparison is insensitive to whether characters of the program elements are uppercase or lowercase.